

**Pub. No.:** 07-319681 [JP 7319681 A ]

**Published:** December 08, 1995 (19951208)

**Inventor:** SUGIURA ATSUSHI

**Applicant:** NEC CORP [000423] (A Japanese Company or Corporation), JP (Japan)

**Application No.:** 06-107332 [JP 94107332]

**Filed:** May 23, 1994 (19940523)

**International Class:** [ 6 ] G06F-009/06; G06F-009/44

**JAPIO Class:** 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

#### **ABSTRACT**

**PURPOSE:** To add a feature amount utilized for generalizing examples even by a user without a programming technique, in programming by the examples.

**CONSTITUTION:** A marking execution part 31 marks objects based on object data stored in an object storage part 21 by using retrieval/replacement visible rules prepared with a retrieval/replacement visible rule editor by the user and stores the identification number of the marked object and the identification number of the rule in a referable object storage part 22. A feature amount generation part 32 generates a new feature amount by using data stored in the referable object storage part 22 and a relation storage part 23 and stores the generated feature amount in a feature amount storage part 24. A code generation part 33 uses the operation history of the user, the feature amount stored in the feature amount storage part 24 and the attribute values of the respective objects stored in the object storage part and generates a program by an inductive learning method.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-319681

(43) 公開日 平成7年(1995)12月8日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	5 3 0 X	7230-5B		
9/44	5 5 0 R	7737-5B		

審査請求 有 請求項の数 3 O L (全 8 頁)

(21) 出願番号 特願平6-107332

(22) 出願日 平成6年(1994)5月23日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 杉浦 淳

東京都港区芝五丁目7番1号 日本電気株式会社内

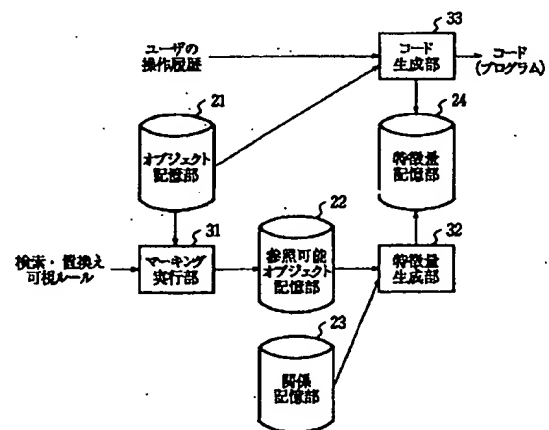
(74) 代理人 弁理士 京本 直樹 (外2名)

(54) 【発明の名称】 例示によるプログラミング方式

(57) 【要約】

【目的】 例示によるプログラミングにおいて、プログラミング技術のないユーザでも、例示の一般化に利用される特徴量を追加することを可能とする。

【構成】 マーキング実行部21はユーザが検索・置換え可視ルールエディタで作成した検索・置換え可視ルールを用いてオブジェクト記憶部21に記憶されたオブジェクトデータに基きオブジェクトのマーキングを行い、マークされたオブジェクトの識別番号とルールの識別番号を参照可能オブジェクト記憶部22に格納する。特徴量生成部32は参照可能オブジェクト記憶部22と関係記憶部23に記憶されているデータを用いて、新たな特徴量を生成し、生成された特徴量を特徴量記憶部24に格納する。コード生成部33は、ユーザの操作履歴と特徴量記憶部24に記憶された特徴量とオブジェクト記憶部に記憶された各オブジェクトの属性値を用いて帰納学習方法によりプログラムを生成する。



#### 【特許請求の範囲】

【請求項1】 グラフィカルエディタ上での例示に基き前記例示に該当する特徴量を選択して自動プログラミングを行うシステムにおける例示によるプログラミング方式において、グラフィカルオブジェクト（以降、単にオブジェクトと称する）の識別番号と前記オブジェクトの属性値などのオブジェクトデータが予め記憶されるオブジェクト記憶部と、予め決められた方法でマーキングされた前記オブジェクトの識別番号と前記オブジェクトのマーキングに使われたルールの識別番号を記憶する参照可能オブジェクト記憶部と、予め前記オブジェクトの属性と前記属性との関係の組が記憶される関係記憶部と、前記特徴量を記憶する特徴量記憶部とを具備し、予め備えたルールエディタで作成した前記ルールを用いて前記オブジェクト記憶部に記憶されたオブジェクトデータに基き前記ルールの適用対象であるオブジェクトのマーキングを行い前記マーキングされたオブジェクトの識別番号と前記マーキングに使われた前記ルールの識別番号を前記参照可能オブジェクト記憶部に格納するマーキング実行部と、前記参照可能オブジェクト記憶部と前記関係記憶部に記憶されているデータを用いて予め決められたアルゴリズムにより新たな特徴量を生成し前記生成された特徴量を前記特徴量記憶部に記憶させる特徴量生成部と、入力された操作履歴と前記特徴量記憶部に記憶された前記特徴量と前記オブジェクト記憶部に記憶された各オブジェクトの属性値を用いてプログラムを生成するコード生成部から構成されることを特徴とする例示によるプログラミング方式。

【請求項2】 前記ルールが、検索・置換え可視ルールであることを特徴とする請求項1記載の例示によるプログラミング方式。

【請求項3】 前記コード生成部が、帰納学習方式を用いて前記プログラムを生成することを特徴とする請求項1記載の例示によるプログラミング方式。

#### 【発明の詳細な説明】

##### 【0001】

【産業上の利用分野】 本発明は、グラフィカルエディタ上での例示によるプログラミングに関し、特に、例示によるプログラミングにおける一般化方式に関する。

##### 【0002】

【従来の技術】 例示によるプログラミングは、Cypherらの文献（“Watch What I Do: Programming by Demonstration”, MIT Press, 1993）にあるように、ユーザがシステム上で行なった操作を記録しておき、該操作を一般化し、システムが実行可能なプログラムに変換することにより、ユーザがプログラミングを行なうことを可能とする方法である。

【0003】 該文献の中（該文献 pp. 341-360）で、Lieberman は、グラフィカルエディタにおいてグラフィカルオブジェクトの編集を行なうためのマクロ作成を、例示

によるプログラミングで行なうシステムについて述べている。該システムにおいては、ある特定のオブジェクト（プログラムの引数、プログラム中で新規に生成されたオブジェクトなど）とその他のオブジェクトの関係を表す特徴量を用いて、一般化を行なう方式を採用している。例えば、図8（a）において、Obj-aをユーザが指定した引数とし、例示として「Obj-1とObj-2を選択する」という操作をユーザがシステムに与え図8（b）の状態にした場合を考える。Lieberman の方式においては、システムは引数であるObj-aと選択されたObj-1とObj-2の相対的な位置関係を考慮し、ユーザが与えた「Obj-1とObj-2を選択する」という操作を「Obj-aより左側にあるオブジェクトを選択する」というように一般化する。この一般化により、グラフィカルエディタ上のオブジェクトの数や種類が異なるような状況でも、ユーザの意図通りに動作するプログラムが生成される。

##### 【0004】

【発明が解決しようとする課題】 上述した文献に示されたような方式を用いて、例示の一般化を行なったとしても、ユーザの意図がシステムにあらかじめ用意された特徴量だけで表現できない場合には、ユーザが望むように一般化することが不可能であった。

【0005】 例えば、図9に示す棒グラフにおいて、ユーザが例示を行ない、システムが一般化する場合について説明する。図9の棒グラフは、矩形、線、及び文字といったグラフィカルオブジェクトより構成されており、ここでのユーザの目的は、3カ月の売上が破線で示された売上高より多い期間を表す文字の背景色を灰色に変更する、という動作を行なうプログラムを記述することとする。

【0006】 この場合、売上が破線を越えている期間は“4-6”月である。ユーザは、グラフィカルオブジェクトの編集機能を用いて、図9（a）に示す初期状態から図9（d）に示す最終状態までの例示を、システムに対して行なう。すなわち、先ず図9（b）に示すように文字オブジェクト“4-6”を選択して、次に図9

（c）に示すように選択された文字オブジェクトの背景色を灰色に変えて、最後に図9（d）に示すように選択状態を解除する。

【0007】 異なる状況（例えば、破線の位置が変わった場合、売上が別の年のものにかわった場合）でも、ユーザが意図通りに動作するプログラムをシステムが生成するためには、システムは、「文字オブジェクト“4-6”を選択する」というユーザの操作に対し、どのような条件を満足するオブジェクトが選択されたのかを見つけたし、一般化を行なわなければならない。例えば、「破線と交わっている矩形と同じ行にある文字を選択する」というような一般化ができなければならない。

【0008】 しかしながら、この例の場合、Lieberman

の方式では、上記のような一般化を行なうことは不可能である。該方式では、特定のオブジェクトとその他のオブジェクトの関係を表す特徴量を用いて一般化を試みる。そのため、上記のような一般化を行うためには、破線と交わっている矩形を図9のグラフィカルオブジェクトの中から検索し、プログラム中で特定のオブジェクトとして指定する必要があるが、該方式では、そのような手段は提供されていないため一般化に必要な特徴量を生成することができない。そのため一般化に失敗し、例えば「文字が“4-6”である文字オブジェクトを選択する」といった、例示を与えた状況でのみ使用できるプログラムを生成することしかできない。

【0009】即ち、ユーザが望む一般化を行なうためには、新たな特徴量を生成する必要があるが、従来はそのためのユーザによるプログラミングが必要であった。

【0010】

【課題を解決するための手段】第1の発明は、グラフィカルエディタ上での例示に基き前記例示に該当する特徴量を選択して自動プログラミングを行うシステムにおける例示によるプログラミング方式において、グラフィカルオブジェクト（以降、単にオブジェクトと称する）の識別番号と前記オブジェクトの属性値などのオブジェクトデータが予め記憶されるオブジェクト記憶部と、予め決められた方法でマーキングされた前記オブジェクトの識別番号と前記オブジェクトのマーキングに使われたルールの識別番号を記憶する参照可能オブジェクト記憶部と、予め前記オブジェクトの属性と前記属性との関係の組が記憶される関係記憶部と、前記特徴量を記憶する特徴量記憶部とを具備し、予め備えたルールエディタで作成した前記ルールを用いて前記オブジェクト記憶部に記憶されたオブジェクトデータに基き前記ルールの適用対象であるオブジェクトのマーキングを行い前記マーキングされたオブジェクトの識別番号と前記マーキングに使われた前記ルールの識別番号を前記参照可能オブジェクト記憶部に格納するマーキング実行部と、前記参照可能オブジェクト記憶部と前記関係記憶部に記憶されているデータを用いて予め決められたアルゴリズムにより新たな特徴量を生成し前記生成された特徴量を前記特徴量記憶部に記憶させる特徴量生成部と、入力された操作履歴と前記特徴量記憶部に記憶された前記特徴量と前記オブジェクト記憶部に記憶された各オブジェクトの属性値を用いてプログラムを生成するコード生成部から構成されることを特徴とする。

【0011】また、第2の発明は、前記ルールが、検索・置換え可視ルールであることを特徴とする。

【0012】さらに、第3の発明は、前記コード生成部が、帰納学習方式を用いて前記プログラムを生成することを特徴とする。

【0013】〔作用〕本発明では、ユーザの意図に関連するオブジェクトを、ユーザがマークできるようにする

ために検索・置換え可視ルールを利用する。該ルールは、グラフィカルオブジェクトの組から形成されるため、CやBASICといったプログラミング言語の知識がないユーザでも、該ルールを作成することが可能である。該ルールによってマークされたグラフィカルオブジェクトは、記憶装置上の参照可能オブジェクト記憶部に記録され、参照可能オブジェクト記憶部に記憶されたグラフィカルオブジェクトとその他のオブジェクトとの何らかの関係を表現する新たな特徴量が自動的に生成されるため、プログラミング技術のないユーザが、新たな特徴量を作成することが可能となる。さらに、該特徴量を用いて例示の一般化が行なわれるため、ユーザが意図した通りの一般化を、システムが行なうことが可能となる。

【0014】

【実施例】次に、本発明について図面を参照して説明する。

【0015】図1は、本発明の一実施例を示すシステム構成図であり、グラフィカルオブジェクトを表示する表示装置2、グラフィカルオブジェクトの操作・編集及び検索・置換え可視ルールの入力するためのマウスおよびキーボードといった入力装置1、例示の一般化に必要な情報を記憶するための記憶装置4、これら表示装置、入力装置及び記憶装置を制御して処理を実行するコンピュータ3より構成される。

【0016】図2は、記憶装置4の内容を示す図であり、記憶装置にはオブジェクト記憶部21、参照可能オブジェクト記憶部22、関係記憶部23、及び、特徴量記憶部24を設ける。オブジェクト記憶部21には、グラフィカルオブジェクトの識別番号と各オブジェクトの属性値がシステムにより記憶される。参照可能オブジェクト記憶部22には、検索・置換え可視ルールによりグラフィカルオブジェクトがマークされた場合に、マークされたグラフィカルオブジェクトの識別番号と該オブジェクトのマークに使われた検索・置換え可視ルールの識別番号がシステムにより記憶される。関係記憶部23には、オブジェクト間の関係を表すために使用される属性と関係の組が記憶される。関係記憶部に記憶されているデータは、システムが予め用意するものである。特徴量記憶部24には、一般化の際に利用可能な特徴量が記憶される。特徴量記憶部に記憶される特徴量には、システムにより予め用意されている特徴量（図2の上から5つ）、および参照可能オブジェクト記憶部と関係記憶部の情報から新たに生成される特徴量（図2では下から10個）がある。

【0017】図3は、本発明の一実施例を示すブロック構成図であり、ユーザが検索・置換え可視ルールエディタで作成した検索・置換え可視ルールを用いてオブジェクト記憶部21に記憶されたオブジェクトデータに基きルールに該当するオブジェクトのマーキングを行い、マ

ークされたオブジェクトの識別番号とマークに使われた検索・置換え可視ルールの識別番号を参照可能オブジェクト記憶部22に格納するマーキング実行部31と、参照可能オブジェクト記憶部22と関係記憶部23に記憶されているデータを用いて、予め定められた方法により新たな特徴量を生成し、生成された特徴量を特徴量記憶部24に格納する特徴量生成部32と、ユーザの操作履歴と特徴量記憶部24に記憶された特徴量とオブジェクト記憶部に記憶された各オブジェクトの属性値を用いて帰納学習方法によりプログラムを生成するコード生成部33から構成される。

【0018】次に、具体例を示す前に、検索・置換え可視ルール及びオブジェクトのマーキングについて説明する。

【0019】検索・置換え可視ルールは、Bellらの文献(“ChemTrains: A Language for Creating Behaving Pictures,” Proceedings of the 1993 IEEE Symposium on Visual Languages, pp.188-195, 1993.)に示されるように、グラフィカルオブジェクトの検索パターンと置換えパターンを有するものであり、該検索パターンと該置換えパターンは、ともにグラフィカルオブジェクトで構成されるものである。検索・置換え可視ルールを任意のグラフィカルオブジェクトの集合に適用すると、該集合中で検索パターンに一致した部分が置換えパターンのオブジェクトに置換えられる。例えば、図5に示すような矩形と破線が交わった部分の検索・置換え可視ルールを、図4(a)に示すグラフィカルオブジェクトの集合に適用すると、矩形と破線が交わった部分が検索され、図4(b)に示すように選択されていなかった矩形が、選択されたノブの付いた状態の矩形に置き換えらる。

【0020】また、本実施例では、上記のように、検索・置換え可視ルールを用いて、選択されていないオブジェクトを、選択されたノブの付いた状態のオブジェクトに置き換えることにより、ユーザがオブジェクトをマークできるものとする。したがって、図4(b)のオブジェクト1はマークされたことになる。

【0021】以下、具体例を用いて説明する。例えば、図4に示す棒グラフにおいて、ユーザが例示を行ない、システムが一般化する場合について説明する。図4の棒グラフは、ユーザがグラフィカルエディタを用いて作成するものであり、矩形、線、及び文字といったグラフィカルオブジェクトより構成されており、ここでのユーザの目的は、3カ月の売上高が破線で示された売上高より多い期間を表す文字の背景の色を灰色に変更する、という動作を行なうプログラムを記述することとする。また、初期状態において、記憶装置は図6に示すように、参照可能オブジェクト記憶部には何も記憶されておらず、特徴量記憶部にも初期の特徴量が記憶されているだけとする。

【0022】「発明が解決しようとする課題」で述べた

ように、図6の特徴量記憶部に記憶されている特徴量のみでは、ユーザが望んだ一般化が行なわれず、ユーザは所望のプログラムを得ることはできない。そこで、ユーザは、検索・置換え可視ルールを用いて、一般化の際に考慮されるべきグラフィカルオブジェクトをマークする。すなわち、ユーザの例示は次のようなものになる。まず、図4(a)のグラフィカルオブジェクトの集合に対して、図5に示す検索・置換え可視ルールを適用する。この結果、上記のように、図4(b)のオブジェクト1がマークされる。次に、図4(c)に示すようにオブジェクト2を選択し、次に図4(d)に示すように選択されたオブジェクト2の背景色を灰色に変え、最後に図4(e)に示すように選択状態を解除する。なお、図5に示す検索・置換え可視ルールはユーザが、グラフィカルエディタと同等のグラフィカルオブジェクト編集機能をもつ検索・置換え可視ルールエディタを用いて作成するものとする。

【0023】グラフィカルオブジェクトがユーザによりマークされると、マーキング実行部31(図3)は、参照可能オブジェクト記憶部22に該オブジェクトとマーキングに利用された検索・置換え可視ルールの識別番号を記憶する。図4の例の場合、参照可能オブジェクト記憶部22には図2に示すように図4に示すオブジェクト1及び図5に示すルール1が記憶される。

【0024】次に、特徴量生成部32は、参照可能オブジェクト記憶部22に記憶されたデータと関係記憶部23に記憶されているデータを組み合わせて新たな特徴量を生成し、生成された特徴量を特徴量記憶部24に記憶する。具体的には、参照可能オブジェクト記憶部22の検索・置換え可視ルール欄に検索・置換え可視ルールVRが記憶されており、関係記憶部の属性欄に属性A、関係欄に関係Relationが記憶されている場合、「VRによりマークされたオブジェクトの属性Aの属性値との間にRelationが成立するか否か」という特徴量が生成される。このように生成された特徴量は、図2に示すように、特徴量記憶部24に記憶される。

【0025】その後で、コード生成部33(図3)は、特徴量記憶部24に記憶されている特徴量とオブジェクト記憶部21に記憶されている各オブジェクトの属性値情報を用いて、ユーザの操作を一般化する。この図4の例の場合、複数の解釈が可能な操作は、図4(c)の「オブジェクト2を選択する」という操作のみであるため、この操作だけ一般化される。

【0026】図4(c)の「オブジェクト2を選択する」という操作は、次のように行われる。ここでは、ユーザが選択したオブジェクト2とそれ以外のオブジェクトを、それぞれ正例、負例とし、正例と負例を完全に区別する条件を、帰納学習方式を用いて生成する。ここで生成される条件は、特徴量記憶部に記憶されている特徴量を論理演算子Vおよび^で組み合わせたものであり、

全ての正例のオブジェクト（この場合、図4（c）で選択されたオブジェクト2）は該条件を満足し、全ての負例のオブジェクト（この場合オブジェクト2以外の全てのオブジェクト）は該条件を満足しないものである。

【0027】例えば、（ルール1によってマークされたオブジェクトの上側のy座標y1より下にある）∧（ルール1によってマークされたオブジェクトの下側のy座標y2より上にある）∧（オブジェクトの種類が文字）という条件が生成される。この条件を満たすオブジェクトは、正例であるオブジェクト2のみである。

【0028】上記一般化の結果と図4のユーザの操作を踏まえて、コード生成部33は、図7に示すように動作するプログラムを生成する。即ち、ステップ1は、検索・置換え可視ルール“ルール1”を用いてマーキングを行うプログラム、ステップ2は、（ルール1によってマークされたオブジェクトのy1より下にある）∧（ルール1によってマークされたオブジェクトのy2より上にある）∧（オブジェクトの種類が文字）という条件を満たすオブジェクトを選択するプログラム、ステップ3は、選択されたオブジェクトの背景色を灰色に変えるプログラム、ステップ4は、全ての選択を解除するプログラムからなる。ステップ1からステップ4は、それぞれ図4（b）から図4（e）の操作に対応して生成される。ステップ1においてマークされるオブジェクトは、プログラムが起動されるその時々状況によって異なる。即ち、図4の破線の位置が変わっても、検索・置換え可視ルール1によって破線と交わっている矩形をマークすることができるため、ユーザは所望の動作をするプログラムを得ることが可能となる。

【0029】なお、検索・置換え可視ルールは、グラフィカルオブジェクトの検索および置換えができるものであれば、任意の方式の検索・置換え可視ルールを用いてもよい。また、グラフィカルオブジェクトのマーキングは、本実施例で示した以外の方法を採用してもよい。さらに、帰納学習方式は、与えられた特徴量を用いて正例と負例を区別する条件を生成するものであれば、任意の方式を用いることができる。

【0030】本方式は、任意の絵（グラフィカルオブジェクトの集合）を描くことができるグラフィカルエディタを用い、グラフィカルオブジェクトの集合から任意のオブジェクトの組を検索して置き換える検索・置換え可視ルールを用いてマーキングを行い、正例と負例を与え

れば、正例と負例を完全に区別する条件を生成する帰納学習方式を用いるため、本方式は、グラフィカルオブジェクトを扱う任意のシステムにおいて、利用することが可能である。

#### 【0031】

【発明の効果】以上説明したように本発明によれば、ユーザが検索・置換え可視ルールを用いて一般化に必要なグラフィカルオブジェクトをシステムに示すと、システムは該オブジェクトを用いて新たな特徴量を生成することができるようにしたことにより、例示に基くユーザ所望の一般化プログラムの生成を、システムが行なうことが可能となる効果がある。

#### 【図面の簡単な説明】

【図1】本発明の一実施例を示すシステム構成図である。

【図2】図1の記憶装置4の内容を示す図である。

【図3】本発明の一実施例を示すブロック構成図である。

【図4】本実施例における検索・置換え可視ルールを用いて例示を行なう際のグラフィカルオブジェクトの状態の変化を示す図である。

【図5】本実施例における検索・置換え可視ルールの一例を示す図である。

【図6】本実施例における記憶装置の初期状態を示す図である。

【図7】本実施例における生成された一般化プログラムの一例を示す図である。

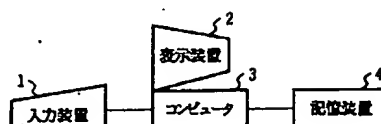
【図8】従来技術の動作説明図である

【図9】従来技術のグラフィカルオブジェクトの状態の変化を示す図である。

#### 【符号の説明】

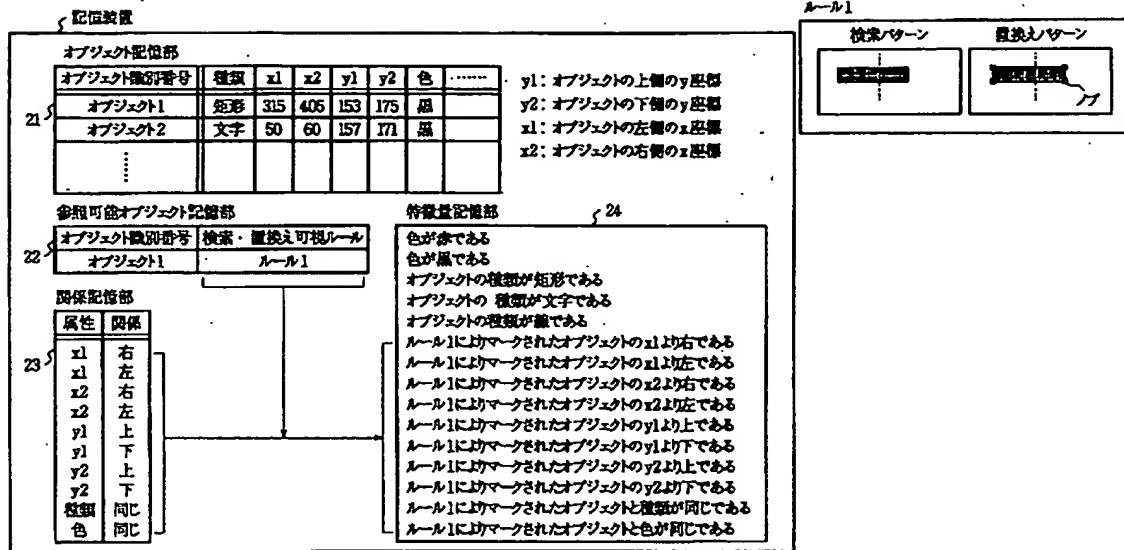
- |    |               |
|----|---------------|
| 1  | 入力装置          |
| 2  | 表示装置          |
| 3  | コンピュータ        |
| 4  | 記憶装置          |
| 21 | オブジェクト記憶部     |
| 22 | 参照可能オブジェクト記憶部 |
| 23 | 関係記憶部         |
| 24 | 特徴量記憶部        |
| 31 | マーキング実行部      |
| 32 | 特徴量生成部        |
| 33 | コード生成部        |

【図1】



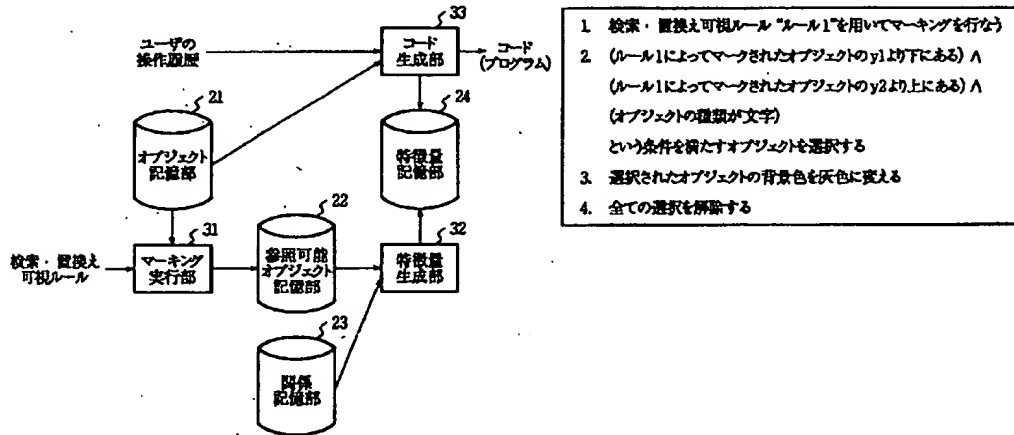
【図2】

【図5】

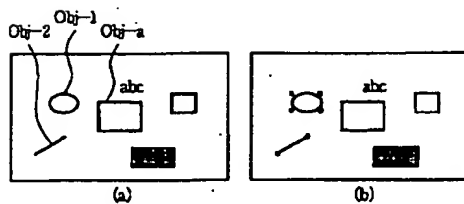


【図3】

【図7】

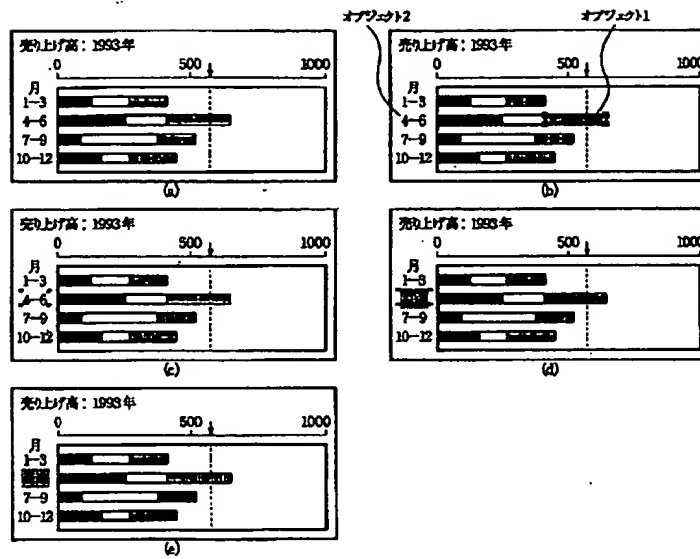


【図8】



BEST AVAILABLE COPY

【図4】



【図6】

# 記憶装置

## オブジェクト記憶部

オブジェクト識別番号	種類	x1	x2	y1	y2	色	.....
オブジェクト1	矩形	315	405	153	175	黒	
オブジェクト2	文字	50	60	157	171	黒	
⋮							

y1: オブジェクトの上側のy座標

y2: オブジェクトの下側のy座標

x1: オブジェクトの左側のx座標

x2: オブジェクトの右側のx座標

## 参照可能オブジェクト記憶部

オブジェクト識別番号	検索・置換え可視ルール

## 特徴量記憶部

色が赤である

色が黒である

オブジェクトの種類が矩形である

オブジェクトの種類が文字である

オブジェクトの種類が線である

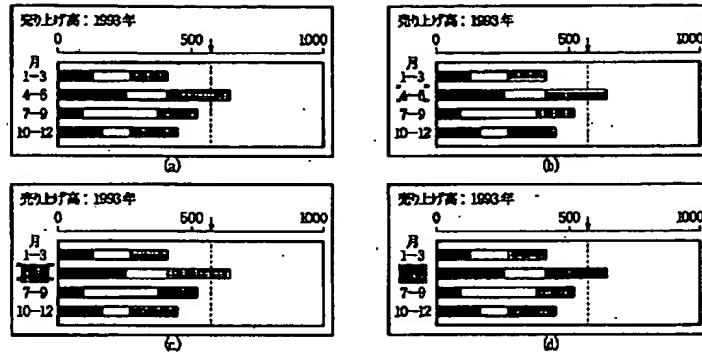
## 関係記憶部

属性	関係
x1	右
x1	左
x2	右
x2	左
y1	上
y1	下
y2	上
y2	下
種類	同じ
色	同じ

BEST AVAILABLE COPY



【図9】



BEST AVAILABLE COPY